

This Page Is Inserted by IFW Operations  
and is not a part of the Official Record

## **BEST AVAILABLE IMAGES**

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

IMAGES ARE BEST AVAILABLE COPY.

**As rescanning documents *will not* correct images,  
please do not report the images to the  
Image Problems Mailbox.**

**THIS PAGE BLANK (USPTO)**

Query/C mmand : prt fu

E6175

---

1 / 1 JAPIO - ©JPO - image

PN - JP 09062624 A 19970307 [JP09062624]  
TI - PROCESSING METHOD AND PROCESSING SYSTEM FOR ON-LINE TRANSACTION  
IN - WATABE SHINICHI; NAKAI KOJI; MIWA NAOKI; SASAKI RIICHIRO  
PA - HITACHI LTD  
AP - JP21846995 19950828 [1995JP-0218469]  
IC1 - G06F-015/00  
IC2 - G06F-015/16  
AB - PROBLEM TO BE SOLVED: To provide the on-line transaction processing system which can continue operation while maintaining a speedy flow of transactions when all nodes are loaded increasingly as the number of transactions increases.  
SOLUTION: At each node, its load rate is monitored, ranked (to L, H1, or H2) by a decision part 13 and mutually updated, and a management table 12 is used for management. A message from an upstream part which is temporarily stored in a common memory is given information on the load state at the upstream part. The process node determination part 14 of the node which has referred to the load state information of the message determines whether this node processes the message by itself or requests another node to process it and sends it through a communication part 17 in the latter case. A node which processes the message determines a processable process pattern by a determination part 16 according to the current load state of itself and controls an operation process part 23.  
COPYRIGHT: (C)1997,JPO

Search statement 3

**THIS PAGE BLANK (USPTO)**

(43)公開日 平成9年(1997)3月7日

### 技術表示箇所

3 1 0 U  
3 8 0 Z

審査請求 未請求 請求項の数12 O L (全 14 頁)

(74)代理人 弁理士 高橋 明夫 (外1名)

**最終頁に続く**

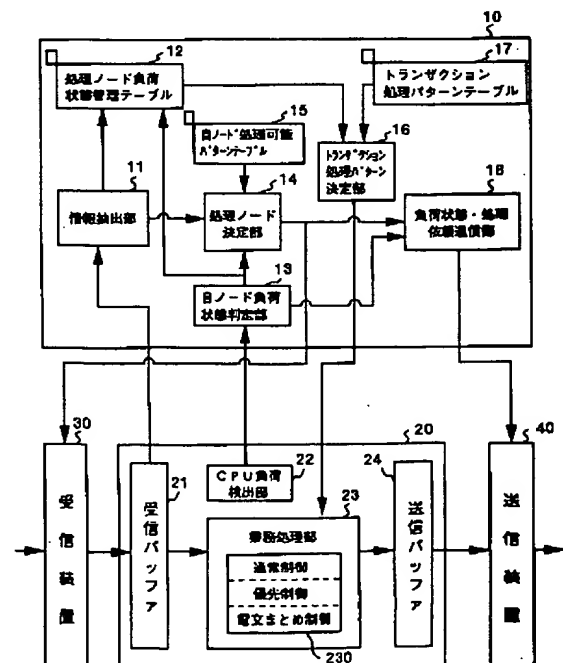
(54) 【発明の名称】 オンラインランザクションの処理方法および処理システム

(57) 【要約】

【目的】トランザクション数の増加で全てのノードの負荷が高くなる場合に、トランザクションの速やかな流れを維持しながら業務を継続できるオンライントランザクションの処理方式を提供する。

【構成】各ノード１～４は、自己の負荷率を監視し判定部１３でランク（Ｌ、Ｈ１、Ｈ２）付けし、相互に更新して管理テーブル１２で管理している。共有メモリ５に一時的に格納される上流からの電文には、上流における負荷状態情報が付帯されている。電文の負荷状態情報を受信装置３０、情報抽出部１１を経由して参照したノードの処理ノード決定１４部は、その電文を自ノードで処理するか他ノードに依頼するか決定し、依頼する場合には通信部１７を経て送信する。電文を処理するノードは、決定部１６でその時点の自負荷状態を基に、処理可能な処理パターンを決定して業務処理部２３を制御する。

☒ 1



**【特許請求の範囲】**

【請求項1】 複数の処理ノード間で負荷分散しながら進めるオンライントランザクションの処理方法において、

自ノードの負荷を監視して負荷状態の程度を判定するとともに、ノード間で前記負荷状態の程度を交信して管理し、上流からトランザクションに相応する電文を受信し、それに付帯されている上流での負荷状態を示す電文の前記負荷状態の程度と前記自ノードの負荷状態の程度を基にして、自ノードでトランザクション処理するか他ノードに依頼処理するか決定することを特徴とするオンライントランザクションの処理方法。

【請求項2】 請求項1において、

前記電文の負荷状態の程度が自ノードのそれより高い場合及び前記自ノードの負荷状態の程度が通常状態より高く両者の程度が同等の場合は、処理可能な他ノードのあるときにはそれへ処理依頼することを特徴とするオンライントランザクションの処理方法。

【請求項3】 請求項1において、

前記電文の負荷状態の程度が通常状態より高く、高い方向への推移が予想できる時間帯の場合は、処理可能な他ノードのあるときにはそれへ処理依頼することを特徴とするオンライントランザクションの処理方法。

【請求項4】 請求項2または3において、

前記処理可能な他ノードは、前記自ノードの負荷状態の程度と同等以下であることを特徴とするオンライントランザクションの処理方法。

【請求項5】 複数の処理ノード間で負荷分散しながら進めるオンライントランザクションの処理方法において、

自ノードの負荷を監視して負荷状態の程度（ランク）を判定し、ノード間で前記負荷状態の程度を交換して管理し、上流からトランザクションに相応する電文を受信したときには前記負荷状態の程度が低いノードが分担するようになり、トランザクション処理を実行するノードは自己の前記負荷状態の程度に応じて処理パターンを選択することを特徴とするオンライントランザクションの処理方法。

【請求項6】 請求項5において、

前記処理パターンは、電文の到達順に処理する通常制御、電文の種別による優先順に処理する優先制御及び複数の電文をまとめて処理する電文まとめ制御を含むことを特徴とするオンライントランザクションの処理方法。

【請求項7】 複数の処理ノード間で負荷分散しながら進めるオンライントランザクションの処理方法において、

自ノードの負荷を監視して負荷状態の程度を判定するとともに、ノード間で前記負荷状態の程度を交換して管理し、上流からトランザクションに相応する電文を受信し、それに付帯されている上流での負荷状態を示す電文

の負荷状態の程度と前記自ノードの負荷状態の程度を基にして、自ノードでトランザクション処理するか他ノードに依頼処理するか決定し、トランザクション処理を実行するノードは自己の前記負荷状態の程度に応じて処理パターンを選択することを特徴とするオンライントランザクションの処理方法。

【請求項8】 受信装置、送信装置、業務処理部を有する複数の処理ノードがネットワークに接続され、ノード間で負荷分散しながら進めるオンライントランザクションの処理システムにおいて、

前記処理ノードは、自ノードの負荷を監視して負荷状態の程度を判定する手段、自ノードの負荷状態の程度と他ノードそれを管理する負荷状態管理手段、トランザクションに相応する電文を受信したときに前記負荷状態管理手段を参照して、自ノードで処理するか他ノードに処理依頼するかを決定する処理ノード決定手段、自ノードでトランザクション処理を実行する場合に、自己の負荷状態の程度に適応した処理パターンを選択し前記制御部を制御する処理パターン決定手段を含む制御手段を備えることを特徴とするオンライントランザクションの処理システム。

【請求項9】 請求項8において、

前記処理ノード決定手段は、前記電文の付帯する上流の前記負荷状態を基に、負荷分散を促進させるように処理ノードを決定することを特徴とするオンライントランザクションの処理システム。

【請求項10】 請求項8または9において、

前記ノード間の前記負荷状態の交換および前記処理依頼は、各ノードによる周期的なブロードキャストで行うことを特徴とするオンライントランザクションの処理システム。

【請求項11】 請求項8または9または10において、

最初に、電文を受信し実行する処理ノードを決定するノードは、システム内で予め定めであることを特徴とするオンライントランザクションの処理システム。

【請求項12】 請求項8または9または10または11において、

電文を受信し実行する処理ノードを決定するノードは、前記処理依頼するノードに転移されることを特徴とするオンライントランザクションの処理システム。

**【発明の詳細な説明】****【0001】**

【産業上の利用分野】 本発明は、分散オンライントランザクションシステムに関し、特に、高トラフィック時のトランザクション処理方式に関する。

**【0002】**

【従来の技術】 複数の計算機をネットワークで接続した分散システムの負荷分散方式として、予め互いに代替する計算機を登録しておき、オンライン分散の可能なネ

ットワークジョブを最も負荷の低い計算機に分担させる、特開昭 62-245361 号に記載の方式、一定期間毎に CPU 負荷率を求め、負荷を均等化するように業務の分担を変更する、特開平 1-283663 号に記載の方式、業務タスクを時刻と共に動的に変更することにより分散システム内の負荷を均等化する方式等が知られている。

#### 【0003】

【発明が解決しようとする課題】複数の計算機がネットワークにより接続された分散オンライントランザクションシステムにおいては、トランザクションは機能分散あるいは負荷分散した複数の計算機が互いに協調しつつ処理を行っている。このようなシステムでは、ある計算機にて発生したトランザクションが他の計算機へと移動するため、トランザクションの発生量がピークとなった場合に、複数の計算機の負荷が同様に増加し、計算機の処理性能を超える事態を招きかねない。また、処理性能を超えなくても CPU 負荷等の利用率が 100% に近くなると、トランザクションの処理性能は著しく劣化する。

【0004】本発明の目的は、トランザクション数の増加によりすべての計算機の負荷が同様に高くなるような場合のトランザクション処理において、トランザクションの速やかな流れを維持できる早期の負荷分散と、処理の継続を確保するオンライントランザクションの処理方法と処理システムを提供することにある。

#### 【0005】

【課題を解決するための手段】本発明の上記した目的は、複数の処理ノード間で負荷分散しながら進めるオンライントランザクションの処理方法において、自ノードの負荷を監視して負荷状態の程度を判定するとともに、ノード間で前記負荷状態の程度を交信して管理し、上流からトランザクションに相応する電文を受信し、それに付帯されている上流での負荷状態を示す電文の前記負荷状態の程度と前記自ノードの負荷状態の程度を基にして、自ノードでトランザクション処理するか他ノードに依頼処理するか決定することにより達成される。

【0006】前記電文の負荷状態の程度が自ノードのそれより高い場合及び前記自ノードの負荷状態の程度が通常状態より高く両者の程度が同等の場合は、処理可能な他ノードのあるときにはそれへ処理依頼することの特徴とする。

【0007】また、前記電文の負荷状態の程度が通常状態より高く、高い方向への推移が予想できる時間帯の場合は、処理可能な他ノードのあるときにはそれへ処理依頼することの特徴とする。

【0008】前記処理可能な他ノードは、前記自ノードの負荷状態の程度と同等以下であることを特徴とする。

【0009】さらに、本発明の目的は、複数の処理ノード間で負荷分散しながら進めるオンライントランザクションの処理方法において、自ノードの負荷を監視して負

荷状態の程度を判定し、ノード間で前記負荷状態の程度を交換して管理し、上流からトランザクションに相応する電文を受信したときには前記負荷状態の程度が低いノードが分担するようになり、トランザクション処理を実行するノードは自己の前記負荷状態の程度に応じて処理パターンを選択することにより達成される。

【0010】前記処理パターンは、電文の到達順に処理する通常制御、電文の種別による優先順に処理する優先制御及び複数の電文をまとめて処理する電文まとめ制御を含むことを特徴とする。

【0011】上記した本発明の方法を適用したシステムは、受信装置、送信装置、業務処理部を有する複数の処理ノードがネットワークに接続されていて、前記処理ノードが、自ノードの負荷を監視して負荷状態の程度を判定する手段、自ノードの負荷状態の程度と他ノードそれを管理する負荷状態管理手段、トランザクションに相応する電文を受信したときに前記負荷状態管理手段を参照して、自ノードで処理するか他ノードに処理依頼するかを決定する処理ノード決定手段、自ノードでトランザクション処理を実行する場合に、自己の負荷状態の程度に適応した処理パターンを選択し前記御無処理部を制御する処理パターン決定手段を含む制御手段を備えることにより実現できる。

#### 【0012】

【作用】本発明によれば、上流からの電文に付帯されている上流での負荷状態を基にして、自ノードでトランザクション処理するか他ノードに依頼処理するか決定するようにしているので、上流での負荷状態がいち早く反映された負荷分散が行われ、システム内の負荷の均等化が早期に実現されるので、トランザクションの速やかな流れを維持して、システムの処理性と信頼性を確保できる。

【0013】また、システム内各処理ノードの負荷状態が全体的に高まる場合にも、処理ノードの負荷状態に応じた処理パターンによって業務が継続されるので、システムの処理性を向上できる。

#### 【0014】

【実施例】以下、本発明の一実施例について図を用いて説明する。

【0015】図 2 は、本発明のオンライントランザクション処理システム（以下、OLTP システムと呼ぶ）が適用される情報伝送ネットワークの構成図である。情報伝送ネットワークでは、上位システム 7 から送られてきた電文を、点線内の本 OLTP システムの共有メモリ 5 に一時的に格納したのち、何れかの処理ノードで業務処理し、処理結果の電文を 1 または複数の下位システム 8、9 へと伝送する。説明を簡単にするために、本 OLTP システムの業務処理の内容は、受信した情報をそのまま送信する情報中継とする。

【0016】本実施例の OLTP システムは、ネットワ

ーク装置6により接続された処理ノード1~4からなる分散システムとして構成され、処理ノード1~4は同じ処理機能を有している。また、本実施例では、1つのトランザクションを1つの電文で表わすものとし、このシステム間の電文形式は図3に示すように、ヘッダー部とデータ部を有し、ヘッダー部には送信元、データ個数、電文の種別または種別によって付与される優先度及び、送信元における当該電文の処理時の負荷状態情報が含まれる。データ部には等サイズに分割された複数のデータスロットと最高尾を示すEOFが含まれる。

【0017】図1は、一実施例による処理ノードの構成を示す機能ブロック図である。各処理ノードは従来とほぼ同様の受信装置30、トランザクション処理部20及び送信装置40と、本発明によって付加されたオンライントランザクション処理の制御部10からなる。

【0018】トランザクション処理部20は受信バッファ21、CPU負荷検出部22、メモリ230を含む業務処理部23及び送信バッファ24から構成されている。受信装置30は、上位システム7から共有メモリ5に格納された電文を、制御部10からの指示によって読み出す。また、システム内の他の処理ノードから送信される電文を受信する。他のノードから電文には、そのノードの現時点の負荷状態情報等が含まれる。さらに、送信先下位システムからのACK信号を受信する。

【0019】受信バッファ21は、受信装置30が共有メモリ5より読み出した上位システムからのシステム間の電文や、他の処理ノードからのノード間の電文を一時的に格納する。CPU負荷検出部22は、トランザクション処理部20の処理を行う図示していないCPUの負荷状態を検出する。

【0020】業務処理部23は、オンライントランザクション処理の制御部10から処理パターンの指示を受けて、メモリ230から処理パターンに対応する制御処理を行う特有の構成を有している。この例は中継処理であるから、受信電文のヘッダー部のみ変更して送信バッファ24に転送し、送信装置40から自ノードの送信可能時期に出力する。そして、処理プログラムで定めるタイムアウト内に中継処理のACK信号が受信されると、当該業務処理が完了する。送信装置26はシステム内における同報通信機能を有し、制御部10から渡された電文をシステム内の処理ノードに定周期でブロードキャストする。

【0021】オンライントランザクション処理の制御部10は、上位システムの電文の負荷状態情報や他ノードの電文から負荷状態情報等を抽出する情報抽出部11、システム内の各処理ノードの負荷状態を管理する処理ノード負荷状態管理テーブル12、自処理ノードの負荷状態を判定する自ノード負荷状態判定部13、上位からの電文を処理するノードを決定する業務処理ノード決定部14、負荷状態に応じ自ノードで処理可能なパターンを

示す自ノード処理可能パターンテーブル15、自ノードにおける業務処理の処理方式を決定するトランザクションの処理パターン決定部16、トランザクションの処理パターンテーブル17、負荷状態・処理依頼通信部18から構成されている。

【0022】次に、オンライントランザクション処理の制御部10の動作を詳細に説明する。この処理動作は、予め任意に設定されるシステム内の入口ノードとそれ以外のノードとで相違する。ここでは、仮に処理ノード1が入口ノードとする。

【0023】図4は、自ノード負荷状態判定フローである。本処理は各ノードに共通で、自ノード負荷状態判定部13によって、定期的（例えば、1秒毎）に処理される。まず、CPU負荷部22により検出されたトランザクション処理部20の現時点の負荷状態を取り込み（s101）、CPU負荷率が所定値（ここでは、40%）未満であるか判定する（s102）。そうであれば（Yes）、処理ノード負荷状態テーブル12の自ノードエリアに、通常負荷=NOと書き込む（s104）。負荷率が40%以上であれば、CPU負荷率が所定値（ここでは、80%）以上であるか判定する（s103）。Noであれば、自ノードエリアに高トラフィック1=H1と書き込む（s105）。Yesであれば、高トラフィック2=H2と書き込む（s106）。最後に、通信部18で作成した電文によって、送信装置40から自己の負荷状態情報をブロードキャストする。図5に、処理ノード負荷状態管理テーブルの記憶状態の一例を示す。

【0024】図6は、他ノードの電文の取り込み処理フローである。本処理は基本的には各ノード共通で、情報抽出部11によって定期的に処理される。受信バッファ21内の他ノード電文の格納領域から電文を読み出す（s201）。ノード間の電文のフォーマットは、図7

(a)に示すように、負荷状態情報と処理依頼情報を含んでいる。次に、負荷状態情報を抽出し（s202）、処理ノード負荷状態テーブル12の該当ノード（当該電文の送信元）のエリアに書き込む（s203）。さらに、自ノードへの処理依頼の有無を判定し（s204）、無ければ終了する。自ノードへの処理依頼があれば、それを処理ノード決定部14に通知し、後述する処理が行われる。なお、s204、205の処理は、入口ノードでは省略可能である。

【0025】図8は、入口ノードにおける全体的な処理フローである。但し、上述した負荷状態の判定と管理のための処理は含まれない。入口ノードとなっている処理ノード1は、まず、共有メモリ5に格納された電文のヘッダー部から負荷状態情報を読み出す（s301）。この情報は、情報抽出部11を経由して処理ノード決定部14に渡される。処理ノード決定部14は、管理テーブル12から自ノードの負荷状態情報も読み出し（s302）、この2つの負荷状態情報から、当該電文の業務処



理が自ノードで処理するか、自ノード処理可能パターンテーブル15を参照して判定する(s303)。

【0026】図9に、自ノード処理可能パターンテーブルの一例を示す。○は自ノードが処理するパターン、△は他ノードが可能であれば処理依頼するパターン、括弧内は時間帯等によって変更されるパターンである。電文の負荷状態が通常負荷(=NO)で、自ノードがNO/H1のときは自ノードで処理する。電文または自ノードがH2、あるいは電文と自ノードが共にH1以上のときは、他ノードが処理可能であれば処理依頼する。括弧付きの状態は、通常は自ノードによる処理が妥当である。しかし、時間帯によっては、電文の負荷状態が今後さらに高まることが予想できる。そのような場合には、可能であれば他ノードに処理依頼して、システム内での負荷均等化を早めに促進する。

【0027】s303で自ノードによる処理可能と判定されれば(Yes)、受信装置30に指示して共有メモリ5から電文の全体を読み出す(s305)。一方、処理ノードが未定であれば(No)、他のノードで処理可能か判定する(s304)。依頼可能な他ノードが無ければ、ステップs305に移行し自ノードで処理する。

【0028】他ノードで処理可能かの判定は、処理ノード負荷状態管理テーブル12を参照し、自ノードの負荷状態と同等または低いノードがある場合、そのノードへ当該電文についての業務処理を依頼する(s306)。依頼可能なノードが複数ある場合は、最も負荷の低いノードを選択し、複数ある場合には所定の順序に従う。なお、他ノードへの処理依頼は、自ノードより負荷状態の低いノードのみとし、処理依頼の回数を減少して処理速度を高めることもできる。ただし、システム内の負荷の均等化を早めるためには、処理依頼を積極的に行って負荷分散の方が望ましく、これによって、急激に負荷が増大する場合のトランザクションの流れをスムーズにできる。

【0029】他ノードへの処理依頼は、ノード間電文のフォーマット(図7)に示したように、処理依頼情報として依頼先ノード番号を書き込む。なお、本システムでは、各ノードが同一の機能を有しているため、他ノードへの処理依頼と同時に、入口ノードとしての処理権を一緒に移転し、入口ノードを固定しないシステム構成とすることが可能である。

【0030】ステップ305または306の処理後に、現時点の自ノード負荷状態情報を読み出す(s307)。これは、自ノードが新たな電文の処理を引き受けたことで、通常は負荷状態が高まるからである。現時点の自ノードの負荷状態は、本処理とは別に、短周期で繰返し実行されている負荷状態判定処理(図4)によって、更新されている。

【0031】この自ノードの負荷状態を基に、図10に示す処理パターンテーブル17を参照して、業務処理部

23における処理パターンを決定する(s308)。

【0032】図11に、処理パターン決定フローを示す。まず、自ノードの負荷状態が通常負荷(=NO)か判定する(s3081)。通常負荷であれば、通常制御の処理パターンに決定する(s3082)。通常負荷でなければ、自ノードの負荷状態が高トラフィック1(=H1)か判定し(s3083)、そうであれば優先制御の処理パターンに決定し(s3084)、そうでなければ高トラフィック2(H2)であるから、電文まとめ制御の処理パターンに決定する(s3085)。

【0033】通常制御とは、受信順に1の電文毎に送信しACKを受信する業務処理の形態である。優先制御とは、電文のヘッダ部にある優先度の高い順に、電文毎に送信しACKを受信する業務処理の形態である。電文まとめ制御とは、所定数の電文をまとめて送信し、その後延长了タイムアウトでACKを受信する業務処理の形態である。

【0034】以上のように、ステップs303~306で、電文の付帯する上流での負荷状態を考慮したシステム内の負荷分散を早めに、あるいは時間帯などに応じて進める。しかし、それによっても、システム内の各ノードの負荷状態が全体的に高まる場合は、ステップs308による自ノードの負荷状態に応じた処理パターンの選択を行うことで、システムの処理性を向上している。

【0035】次に、入口ノード以外他ノードにおける処理動作を説明する。図12は、他のノードの全体な処理フローである。他ノードの動作も、入口ノード処理権に属する処理ノード決定処理を除けば、入口ノードと同じになる。

【0036】他ノード(例えば、処理ノード2)では、情報抽出部11を介して処理ノード決定部14が、入口ノードからの電文に自ノード宛の処理依頼があるか判定する(s401)。処理依頼があれば、受信装置30に指示して共有メモリ5から電文の全体を読み出す(s402)。以後のs403~s405は、入口ノードにおけるステップs307~s309と同じ処理になる。業務処理後はステップs401に戻り、もし、入口ノードからの処理依頼が無ければ、自ノードの受信バッファ21に蓄積されている未処理の電文に対して、自負荷状態に応じた処理パターンを決定し、業務処理を継続する。

【0037】以上、本実施例によるオンライントランザクション処理システム(OLTPシステム)の構成と動作を詳細に説明した。図13は、情報伝送ネットワークに適用された本システムの情報中継の業務処理を示す説明図である。同図(a)に、時間帯による負荷率の推移、同図(b)に、負荷状態に応じ選択された処理パターンによる業務処理のタイムチャート、同図(c)に、電文種別とそれに応じた優先度及び到達順序のテーブルを示している。

【0038】(a)の負荷率の推移は、1つのCPU負

荷率を示しているが、上流からの電文のもつ負荷状態を反映したものである。ある時間に(12:00～)イベント的に負荷率が急上昇する場合がよくある。この場合、本システム内で各ノードの負荷状態がH2となり、(b)－(ロ)のように電文まとめ処理が行われる。ただし、優先度Aの重要電文(1)については優先制御を行い、他の普通電文(2)、(3)は電文をまとめて送信し、その後にACK(点線で示す)を受信する。また、時間帯(12:15～)によっては常に負荷状態H1状態となることがあり、この場合システム内各ノードの負荷状態はH1以上となり、(b)－(ハ)のように、優先度順による制御が行われる。

【0039】このように、本システムでは電文の負荷状態による負荷の均等化をはかるとともに、通常処理が困難であれば優先制御、さらには電文まとめ制御による業務処理を行うようにしているので、オンライントランザクション処理における負荷の上昇に早期に対応して負荷分散でき、且つ、処理ノード全体の負荷が高まる場合にも業務処理を継続して、システムの処理性を向上している。

【0040】なお、上記の実施例では各ノードの機能が同じとなる負荷分散システムを示したがこれに限られるものではない。OLTPシステム内の複数の処理ノード間で、各々の業務処理に上流、下流の関係が生じるような、機能分散システムにおいても適用可能である。また、上記実施例では、システム内に電文を一時記憶する共有メモリを設けたが、各ノードに補助記憶装置を設けて、まず、入口ノードで一時記憶し、他ノードが処理依頼を受けたときは入口ノードの補助記憶装置からDMAで読み出すようにしてもよい。この場合、入口ノードが他に転移することも可能である。

#### 【0041】

【発明の効果】本発明のオンライントランザクションの処理方式によれば、上流からの電文に付帯される上流での負荷状態を考慮して負荷分散を行うので、システム内の負荷の均等化が早期に実現されるので、トランザクションの速やかな流れを維持して、システムの処理性と信頼性を確保できる効果がある。

【0042】また、システム内各処理ノードの負荷状態

が全体的に高まる場合にも、処理ノードの負荷状態に応じて処理パターンによって業務が継続されるので、システムの処理性を向上できる効果がある。

#### 【図面の簡単な説明】

【図1】本発明の一実施例による、OLTPシステムの処理ノードの構成図。

【図2】本発明のOLTPシステムを適用する情報ネットワークの構成図。

【図3】システム間電文のフォーマットを示す説明図。

【図4】処理ノードの負荷状態の監視・判定処理を示すフローチャート。

【図5】処理ノード負荷状態管理テーブルの説明図。

【図6】ノード間電文の管理処理を示すフローチャート。

【図7】ノード間電文のフォーマットと、各ノードからのブロードキャストのタイミングを示す説明図。

【図8】入口ノードの全体処理を示すフローチャート。

【図9】自ノード処理可能パターンテーブルの説明図。

【図10】トランザクション処理パターンテーブルの説明図。

【図11】処理パターン決定処理を示すフローチャート。

【図12】他ノード(入口ノード以外)の全体処理を示すフローチャート。

【図13】時間推移する情報中継業務への本実施例の適用状況を示す説明図。

#### 【符号の説明】

1～4…処理ノード、5…共有メモリ、7…上位システム、8、9…下位システム、10…オンライントランザクション処理の制御部、11…情報抽出部、12…処理ノード負荷状態管理テーブル、13…負荷状態判定部、14…処理ノード決定部、15…自ノード処理可能パターンテーブル、16…処理パターン決定部、17…トランザクションの処理パターンテーブル、18…負荷状態・処理依頼通信部、20…トランザクション処理部20、21…受信バッファ21、22…CPU負荷検出部、23…業務処理部、230…メモリ、24…送信バッファ、30…受信装置、40…送信装置。

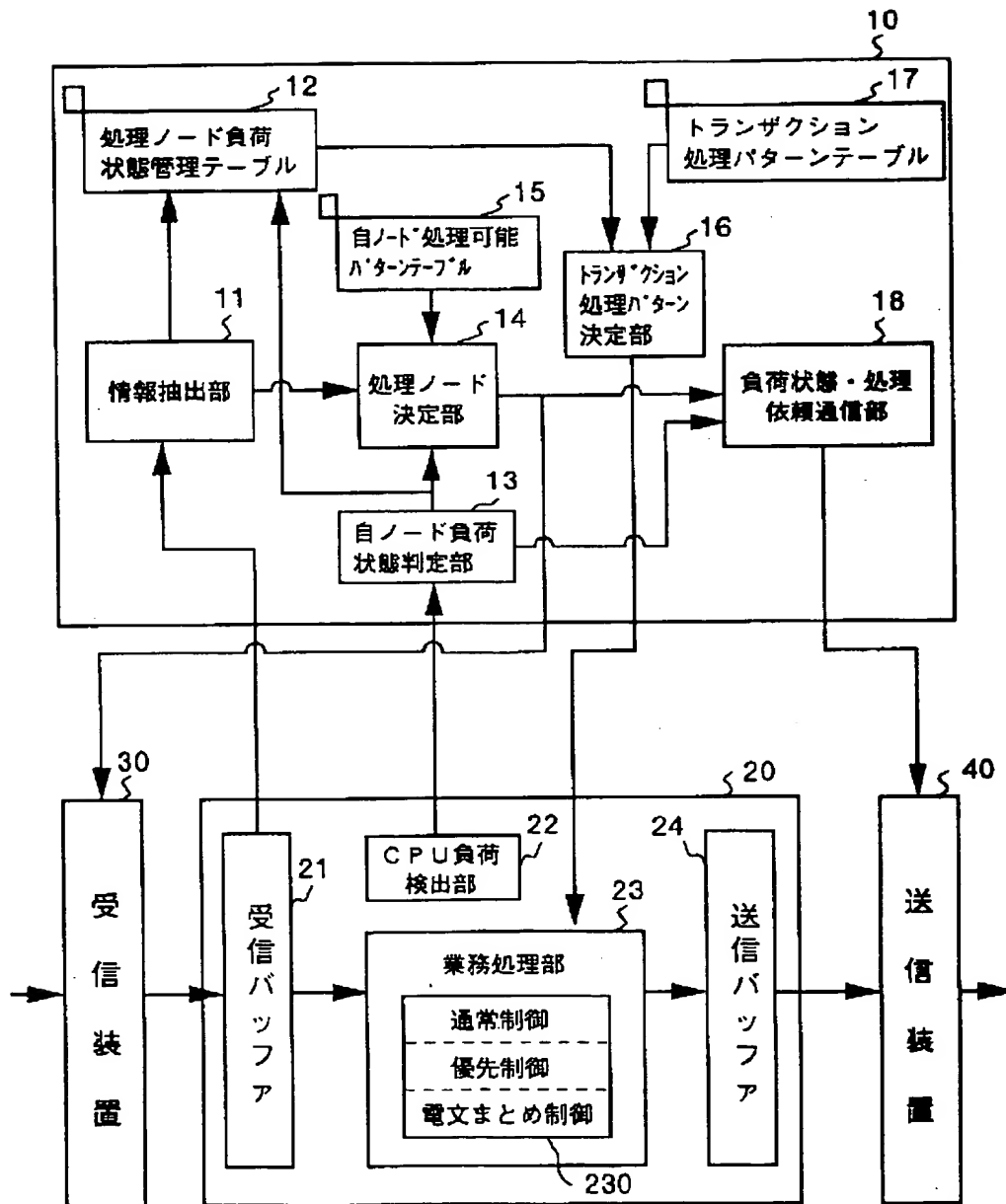
【図10】

図 10

負荷状態	業務処理部の処理パターン
通常負荷	通常制御
高トラフィック1 (定常的)	優先制御
高トラフィック2 (イベント)	電文まとめ制御

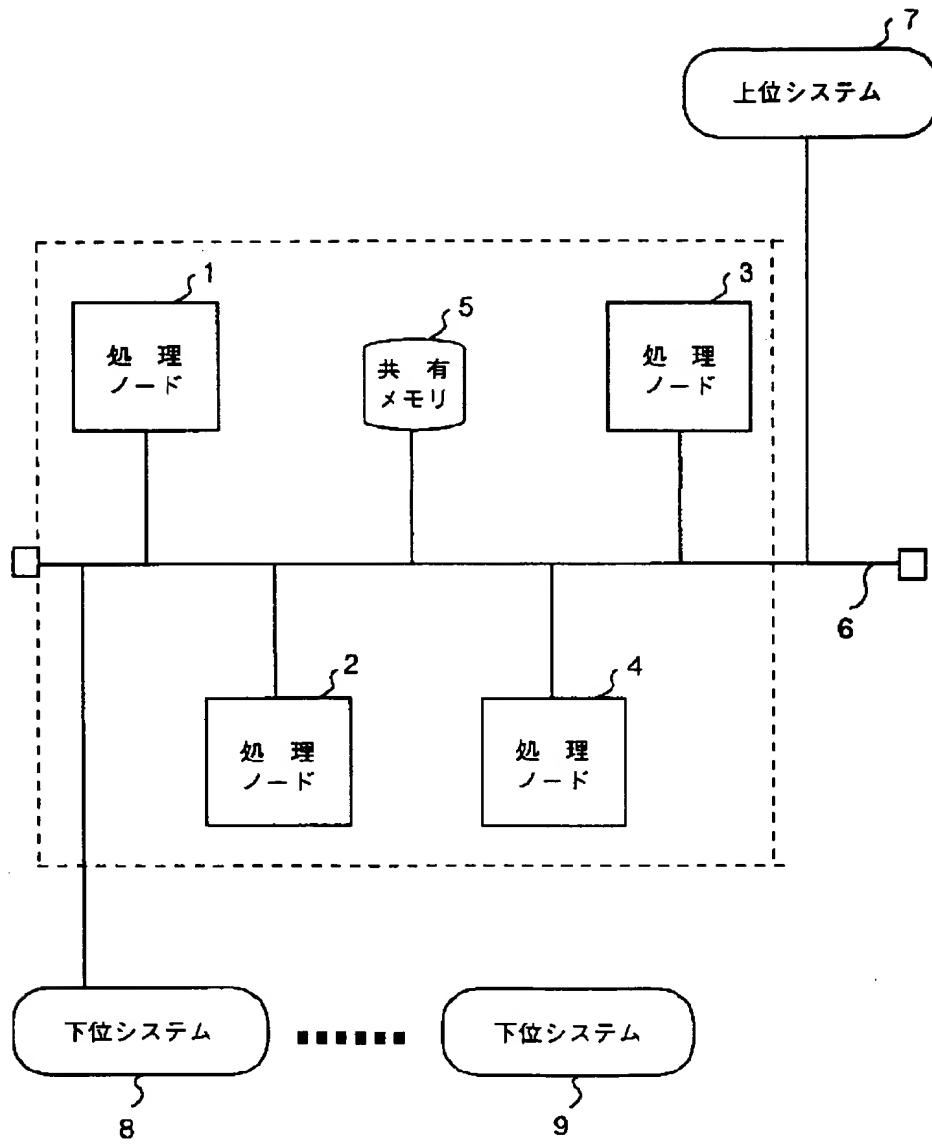
【図1】

図 1



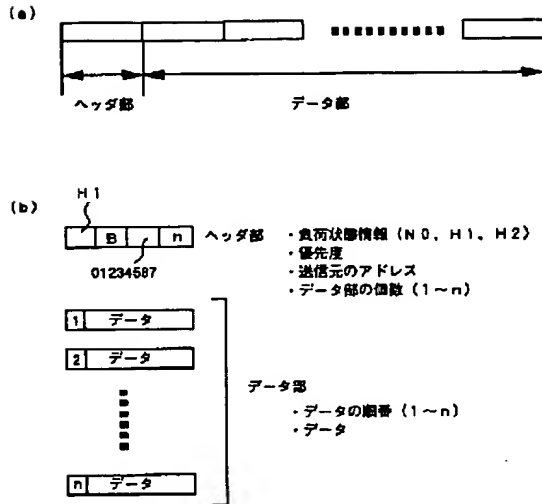
【図2】

図 2



【図 3】

図 3



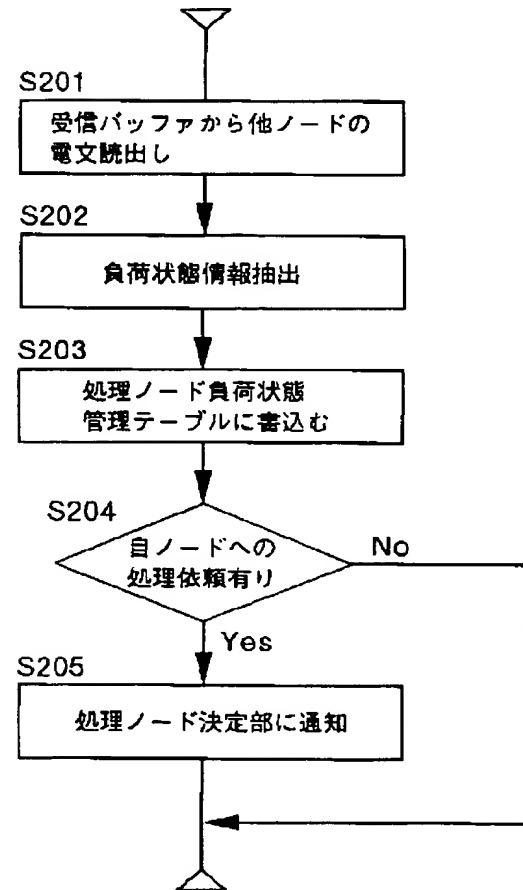
【図 5】

図 5

処理ノード名	負荷状態
処理ノード (1)	通常負荷 (N0)
処理ノード (2)	高トラフィック1 (H1)
処理ノード (3)	通常負荷 (N0)
処理ノード (4)	通常負荷 (N0)

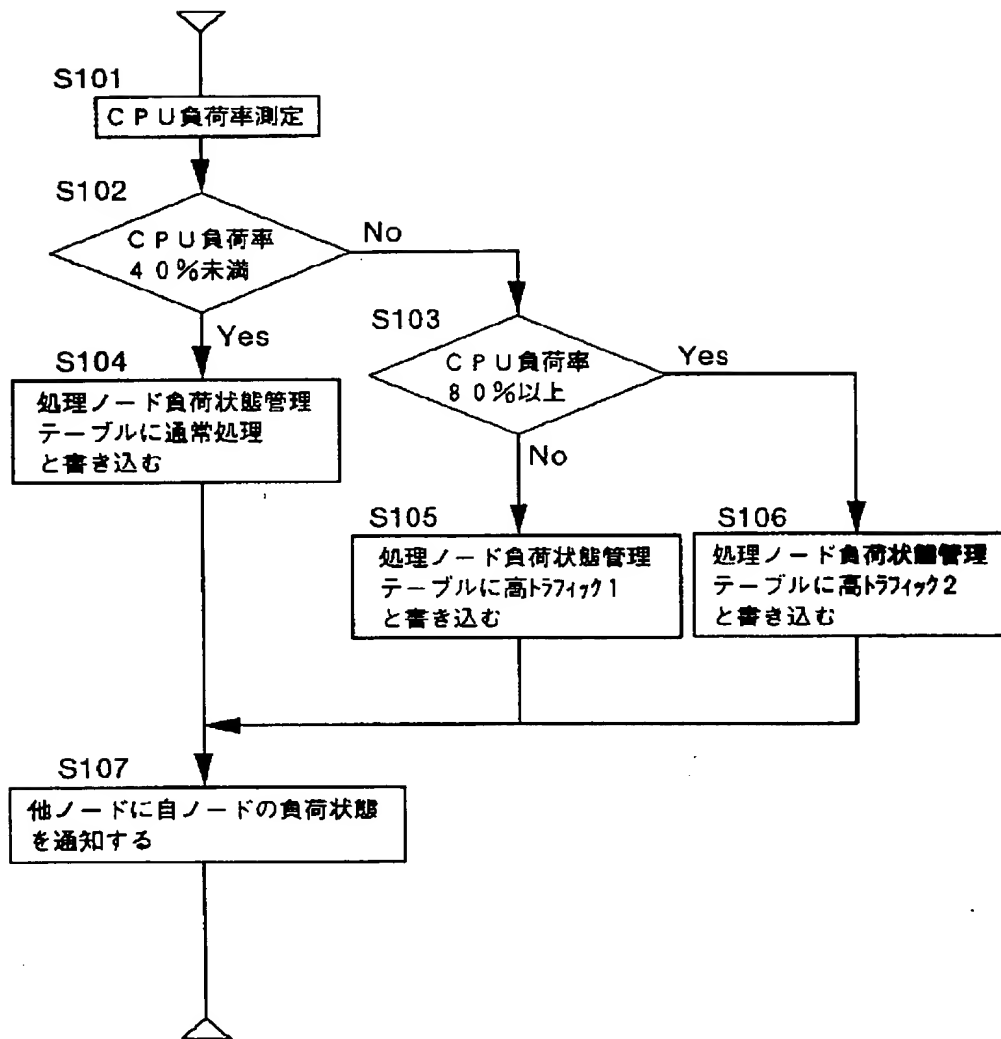
【図 6】

図 6



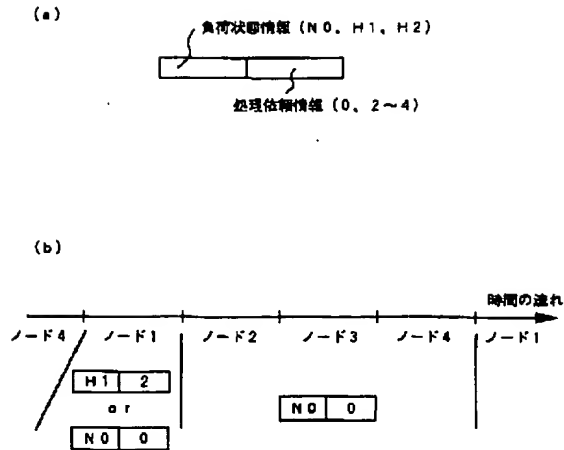
【図 4】

図 4



【図 7】

図 7



【図 9】

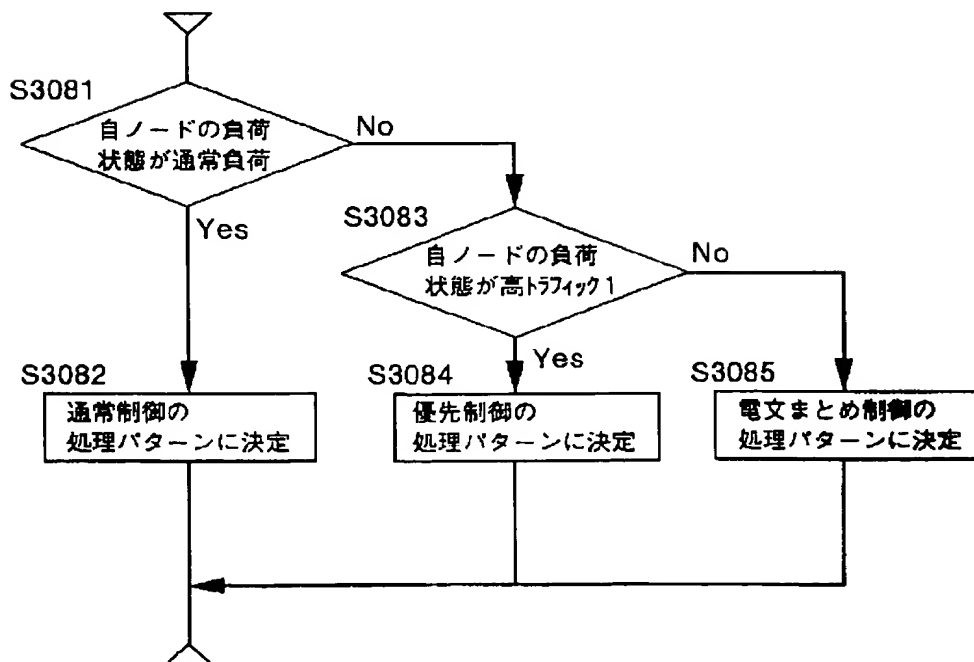
図 9

電文の負荷状態 自ノードの負荷状態	通常負荷	高トラフィック1 (H1)	高トラフィック2 (H2)
通常負荷 (N0)	○	○(△)	△
高トラフィック1 (H1)	○	△	△
高トラフィック2 (H2)	△	△	△

○ 自ノード処理  
 △ 他ノードで可能ならば依頼  
 ( ) 時間帯等によって切替

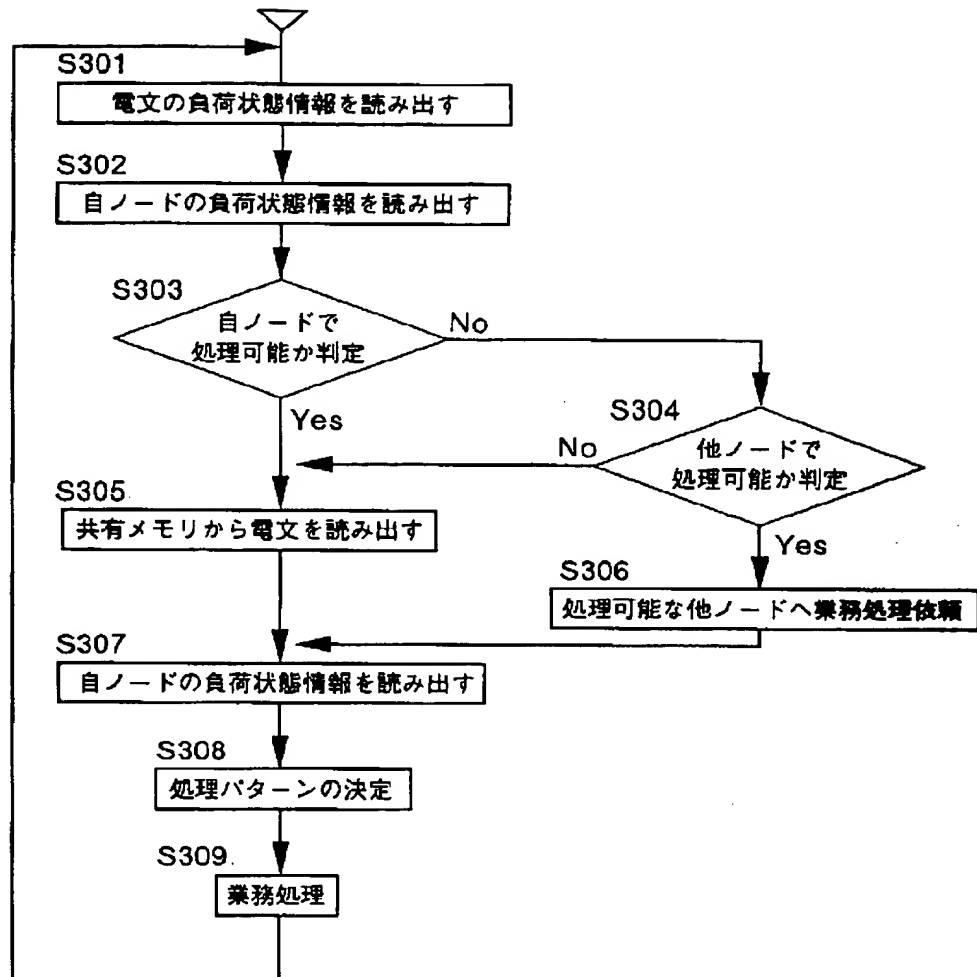
【図 11】

図 11



【図 8】

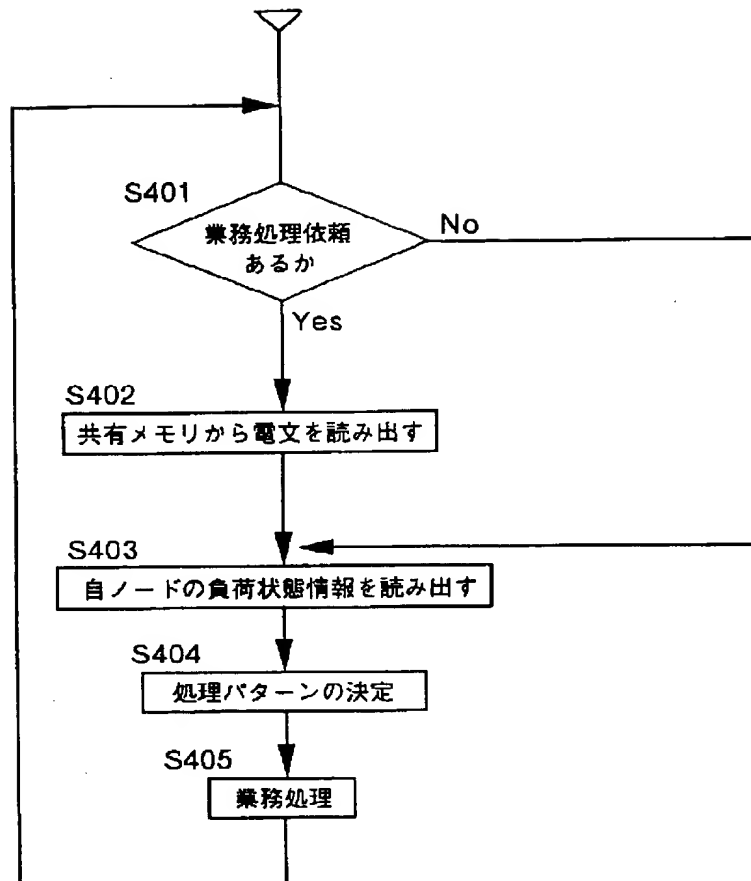
図 8





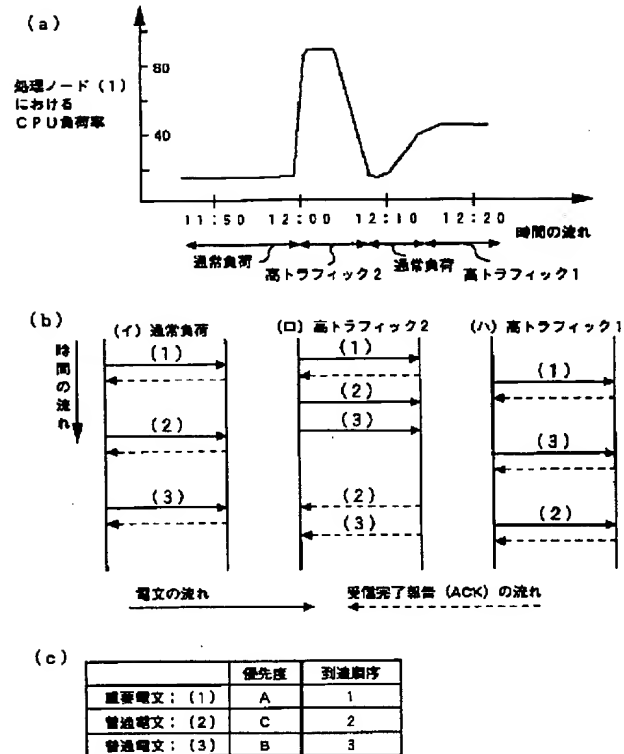
【図 1 2】

図 1 2



【図13】

図 13



フロントページの続き

(72)発明者 佐々木 利一郎  
 茨城県日立市大みか町五丁目2番1号 株  
 式会社日立製作所大みか工場内